



## Eski Kitaplar (Ancient Books)

İran ulusal kütüphanesi Tahran'dadır. Bu kütüphanenin ana hazinesi, bir sıra halinde soldan sağa doğru 0'dan  $n - 1$  'e kadar numaralandırılmış  $n$  masadan oluşur. Her bir masada eski bir elyazması kitap bulunur. Bu kitaplar, yaşlarına göre sıralanmıştır. Kitap isimlerine göre arama yapmak isteyen ziyaretçiler için bu sıralama zorluk çıkarttığından, kütüphane müdürü kitapları isimlerine göre alfabetik olarak sıralamaya karar verir.

Kütüphaneci Aryan bu iş için görevlendirilmiştir. Aryan 0 ile  $n - 1$  arasında farklı tam sayılardan oluşan  $n$  uzunluğunda  $p$  listesi oluşturur. Bu liste, kitapları alfabetik sıraya koymak için gerekli değişiklikleri tanımlar: tüm  $0 \leq i < n$  için, şuanda  $i$  masasında olan kitap  $p[i]$  masasına taşınmalıdır.

Aryan kitapları sıralamaya  $s$  masasından başlar. İşini tamamladıktan sonra aynı masaya dönmek istemektedir. Kitaplar oldukça değerli olduğundan, aynı anda birden fazla kitap taşıyamaz. Kitapları sıralarken Aryan bir faaliyet sırası uygular. Her bir faaliyet aşağıdakilerden biri olmak zorundadır:

- Eğer Aryan bir kitap taşımıyorsa ve bulunduğu masada bir kitap varsa, bu kitabı alabilir.
- Eğer Aryan bir kitap taşıyorsa ve bulunduğu masada başka bir kitap varsa, masadaki kitapla taşıdığı kitabı değiştirebilir.
- Eğer Aryan bir kitap taşıyorsa ve bulunduğu masa boşsa, taşıdığı kitabı masaya bırakabilir.
- Aryan herhangi bir masaya gidebilir. Yürürken tek bir kitap taşıyabilir.

Tüm  $0 \leq i, j \leq n - 1$  için,  $i$  ve  $j$  masaları arasındaki uzaklık tam olarak  $|j - i|$  metredir. Sizden kitapları sıralaması için Aryan'ın yürümesi gereken minimum toplam uzaklığı hesaplamanız istenmektedir.

## Gerçekleştirim detayları

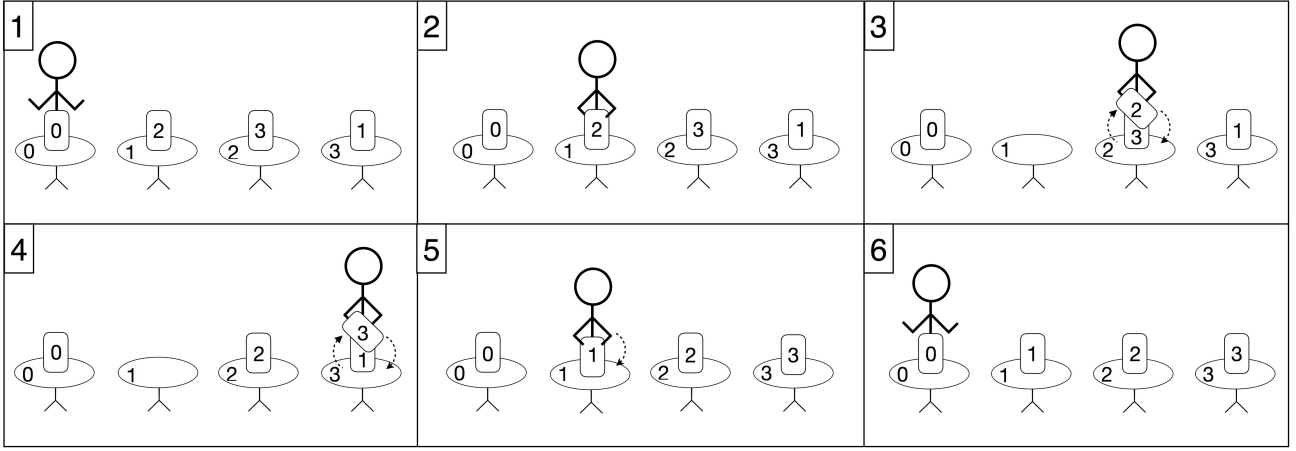
Aşağıdaki prosedürü kodlamalısınız:

```
int64 minimum_walk(int[] p, int s)
```

- $p$ :  $n$  uzunluğunda bir dizi. Başlangıçta  $i$  masasındaki kitap Aryan tarafından alınmalı ve  $p[i]$  masasına bırakılmalıdır (tüm  $0 \leq i < n$  için).
- $s$ : Aryan'ın başlangıçta bulunduğu ve kitapları sıraladıktan sonra da bulunması gereken masanın etiketi.
- Bu prosedür, kitapların sıralanması için Aryan'ın yürümesi gereken en küçük toplam uzunluğu (metre cinsinden) dönmelidir.

## Örnek

```
minimum_walk([0, 2, 3, 1], 0)
```



Bu örnekte  $n = 4$  ve başlangıçta Aryan 0 masasındadır. Kitapları şu şekilde sıralar:

- Aryan, masa 1'e gider ve bu masanın üzerindeki kitabı alır. Bu kitap masa 2'ye bırakılmalıdır.
- Daha sonra, masa 2'ye gider ve taşıdığı kitap ile masanın üzerindekiğini değiştirir. Taşıdığı yeni kitap masa 3'e bırakılmalıdır.
- Daha sonra, masa 3'e gider ve taşıdığı kitap ile masanın üzerindekiğini değiştirir. Taşıdığı yeni kitap masa 1'e bırakılmalıdır.
- Daha sonra, masa 1'e gider ve taşıdığı kitabı masaya bırakır.
- Son olarak, masa 0'a gider.

Not: masa 0'ın üzerindeki kitap zaten olması gereken doğru yerdedir ( masa 0). Yani, Aryan bunu almak zorunda değildir. Bu çözümde yürünen toplam uzunluk 6 metredir. Bu optimal çözümdür. Dolayısıyla, prosedür 6 dönmelidir.

## Sınırlar

- $1 \leq n \leq 1\,000\,000$
- $0 \leq s \leq n - 1$
- $p$  dizisi 0 ve  $n - 1$  arasında  $n$  farklı tam sayı içerir (sınırlar dahildir).

## Altgörevler

1. (12 puan)  $n \leq 4$  and  $s = 0$
2. (10 puan)  $n \leq 1000$  and  $s = 0$
3. (28 puan)  $s = 0$
4. (20 puan)  $n \leq 1000$
5. (30 puan) ek sınır yoktur.

## Örnek değerlendirici

Örnek değerlendirici girdiyi aşağıdaki formatta okur:

- satır 1:  $n$   $s$
- satır 2:  $p[0]$   $p[1]$  ...  $p[n - 1]$

Örnek değerlendirici, `minimum_walk` 'in dönme (return) değerini içeren tek bir satır yazar.